



User's Manual 2009 v 1.0



CONTENTS

1. ABOUT THIS USER MANUAL	1
2. SOFTWARE INSTALLATION	1
2.1 INTRODUCTION	1
2.2 TECHNICAL DATA	2
2.3 SALSA SETUP	2
3. GETTING STARTED - SALSA MAIN SCREEN	2
3.1 SALSA – OFFLINE	3
3.1.1 LOADING THE TRAINING DATA FILE IN THE SPECIFIED FORMAT	3
3.1.1.1 Format Specification	3
3.1.1.2 Loading a training file	4
3.1.2 CHOOSING THE CLASSIFICATION ALGORITHM - QUANTITATIVE DESCRIPTORS	5
3.1.3 LOGIC OPERATORS AND THE EXIGENCY LEVEL	6
3.1.3.1 Logic operators	6
3.1.3.2 Exigency Level	6
3.1.4 CLASSIFICATION MODE	6
3.1.4.1 Self – Learning	7
3.1.4.2 Active Supervised Learning	7
3.1.4.3 Pattern Recognition	8
3.1.5 DATA PROCESSING, TREATMENT AND VISUALIZATION	8
3.1.6 CLASSIFICATION RESULTS AND VISUALIZATION	10
3.1.6.1 Resulting classes - characteristics	11
3.1.6.2 Recognition Results	12
3.1.6.3 Classification Analysis	13
3.1.7 MAPPING CLASSES TO STATES	14
3.2 SALSA – ONLINE	16
3.2.1 ONLINE CONFIGURATION	17
3.2.2 ONLINE CONNECTION	17
3.2.3 RECEIVING ONLINE DATA	19
3.2.3.1 Visualizing incoming data	19
3.2.3.2 Visualizing the Current State	19
4. APPENDIX	20
4.1 ONLINE DATA GENERATOR	20
5. REFERENCES	21

FIGURES TABLE

Figure 1 - SALSA - Main Screen: Options	2
Figure 2 – SALSA - OFFLINE Main Screen	3
Figure 3 – Example of the data (.dat) file	4
Figure 4 – New Context & New Population buttons	4
Figure 5 - Loading a New Context: Induce or Retrieve.....	5
Figure 6 – Loading a ‘New’ population.....	5
Figure 7 - View/Edit File & Imposed Classification Windows.....	5
Figure 8 - Presence Function selection	5
Figure 9 – Connectives & Exigency selection	6
Figure 10 – Classification Mode selection	6
Figure 11 – Self-Learning selectors	7
Figure 12 - Automatic Self-Learning Results: Classes & Partition Quality.....	7
Figure 13 – Supervised Learning Options window	8
Figure 14 – Recognition Options window	8
Figure 15 - Quantitative & Qualitative Descriptor windows	9
Figure 16 – Context Description window	9
Figure 17 – Active quantitative descriptors graph	10
Figure 18 – Active qualitative descriptors graph.....	10
Figure 19 – Application ready for classification	10
Figure 20 – Classification Results using Supervised Learning	11
Figure 21 – Classification visualizing buttons	11
Figure 22 – Profile for each Class	12
Figure 23 – Class Details window	12
Figure 24 - Recognition Results Window	13
Figure 25 - Reference Classification Graph	13
Figure 26 - Analyse Window.....	14
Figure 27 - Membership Graph	14
Figure 28 - Mapping Classes to States Window	15
Figure 29 - Creating the list of States	15
Figure 30 - Assignment of Class to State	15
Figure 31 - View/Edit Mapping & States buttons.....	16
Figure 32 - Current States Graph.....	16
Figure 33 - SALSA Online Main Screen	16
Figure 34 - SALSA Online Configuration	17
Figure 35 – List of Online Quantitative Descriptors	17
Figure 36 – Ready to Send Online configuration.....	18
Figure 37 – Ready to Receive Online data.....	18
Figure 38 – Path for data transfer	18
Figure 39 – Format for the online data file	18
Figure 40 – Format for the output file by SALSA.....	18
Figure 41 – Stop receiving data	19
Figure 42 - Online Quantitative Descriptors.....	19
Figure 43 - Online Qualitative Descriptors	19
Figure 44 - Bar Graph with States	20
Figure 45 - Table with change of State occurrences	20
Figure 46 - Online Data Generator setup window	20
Figure 47 - Online Data generator Main Screen.....	21

1. ABOUT THIS USER MANUAL

This manual is a reference for the qualitative situation assessment of a given industrial process. It provides the guidelines for the construction of classification systems for the offline characterisation and online identification of the functional states of a process by means of a learning classification technique. It combines qualitative and/or quantitative information of the process variables and the expert's knowledge (Kempowsky *et al.*, 2002), (Kempowsky *et al.*, 2003).

The manual is divided into two main chapters and an appendix as follows:

Chapter 2: Covers some technical information, the installation of the application and the system requirements.

Chapter 3: This section provides sufficient information to use the application. It shows an overview of the basic features of SALSA as well as the steps that must be followed.

Appendix: Correspond to the installation of an online data emulator – SALSA Online data r generator.

SALSA application as well as this user manual was developed by the DISCO group from the LAAS-CNRS. All rights reserved.

For further information and details please contact: Joseph AGUILAR-MARTIN (aguilar@laas.fr), Marie-Véronique LE LANN (mvlelann@laas.fr), Tatiana KEMPOWSKY (kempowsky@laas.fr).

This is a 2009 version 1.0 of SALSA User's Manual. Updates are constantly made.

2. SOFTWARE INSTALLATION

2.1 INTRODUCTION

The main function of SALSA is to identify in a qualitative way the current functional state of a supervised process using the online symbolic and/or numeric information from the process variables. To do this SALSA has two main stages. The first, offline, corresponds to a *learning stage* which, from a given data set, generates classes and assigns significant functional states to these classes by a constant dialogue with the expert. The learning stage includes:

- **Class Generation:** a clustering classifier with the capacity to perform a supervised or unsupervised learning procedure. Interaction with the expert is necessary in order to tune up the classifier parameters.
- **Mapping Classes to States:** a dialog tool with the expert, for the assignment of classes, or groups of classes, to meaningful functional states.

The online stage is the *recognition stage*, which will determine the functional state of the process from online data of sensors and actuators or other information from the process variables.

During plant operation SALSA generates the current functional state of the supervised process by using real-time (online) data.

For further information concerning SALSA toolbox or LAMDA classification algorithm the user may refer to (Kempowsky, 2004), (Aguilar-Martin and López de Mántaras, 1982).

2.2 TECHNICAL DATA

- This version of SALSA was programmed in plain ANSI-C, although the underlying philosophy is object-oriented.
- SALSA is a standalone tool, it can operate on its own without need of another software tool.
- The platform for the application's development was LabWindows/CVI (latest version used 8):
- Graphic User Interface
- Source Code programming
- The software may be installed in Windows 98/2000/XP/Vista

2.3 SALSA SETUP

1. From the installation Folder InstallSALSA09_Beta double click at the '**setup.exe**' file and let you guide by the installation wizard.
2. Once SALSA application has been installed verify that the directory **SALSA09v1** has been created.
3. Verify that the sub-folder **\DataTr** has been created in the **SALSA09v1** directory. This folder is required for the online data transfer between SALSA and another application running online.

3. GETTING STARTED - SALSA MAIN SCREEN

When SALSA is launched, Figure 1 shows its principal window. There are three main options to choose:

OFFLINE: Learning Stage. This concerns the design and construction of a classification system, from a set of experimental data. It consists of two main stages: a learning stage (unsupervised or supervised) and the assignment of the resulting classes into significant functional states. Active participation of the process expert is required.

ONLINE: Recognition Stage. Using a previously generated classification system, the current functional state of the process will be identified.

QUIT: This option exits the toolbox. To exit the application the user must first close (quit) the option he is working on (one of the above). Then exit by clicking the QUIT button in the principal MENU window.

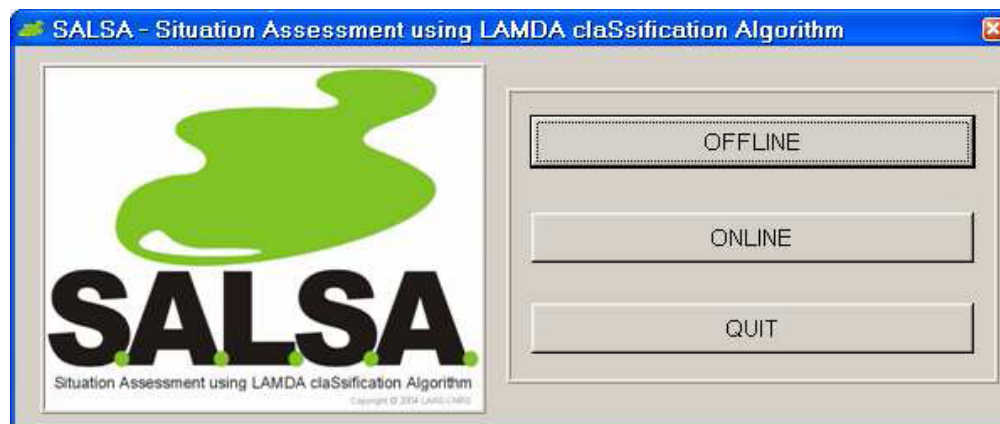


FIGURE 1 - SALSA - MAIN SCREEN: OPTIONS

3.1 SALSA – OFFLINE

Figure 2 shows the main screen for SALSA offline (learning) stage. The following sections give some guidelines of the features and use of this stage.

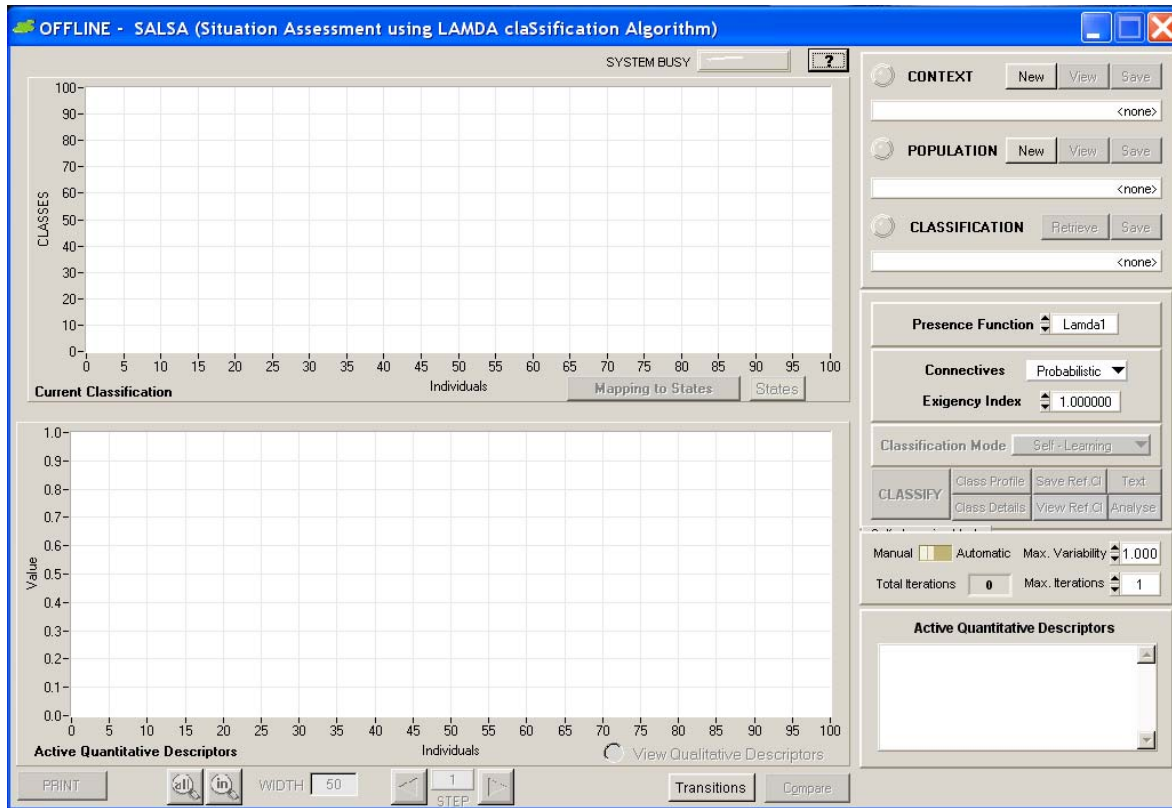


FIGURE 2 – SALSA - OFFLINE MAIN SCREEN

3.1.1 Loading the training data file in the specified format

3.1.1.1 Format Specification

The format for the input (training) files (the context data base and the population to be classified) is a text file with a **.DAT** extension. Lines can be blank ones, comments (MatLab style, always preceded by “%”) or observations. These observations or **individuals** present values for the *context descriptors* in columns separated by ‘blank spaces’ or ‘tabs’ and may have an additional field if we are interested in supervised learning (i.e. the additional field correspond to the pre-defined class).

- **Type of Data:** a descriptor is assumed to be **quantitative** if it starts with a number, a point or a minus sign, and **qualitative** otherwise.

- **The name or tag:** for each descriptor must be a line preceded by the “&” sign and they should be also separated by a ‘blank space’ or ‘tab’.

- **Pre-defined classes:** the additional field for supervised learning will indicate the class where the individual is to be assigned. The algorithm expects these numbers to appear in a semi-ordered way, that is, class “1” has to appear at least once before class “2”; there has to be at least one “2” before any “3” and so on. When these numbers are greater than 0, they create a class. The zero (0) means that the

individual is part of a population that later will undergo pattern recognition (for the moment no class is assigned to this element). Figure 3 shows an example of the training file format for the case of supervised learning. Notice that no name has to be specified for the class column in the tag line.

```
% LEVEL.DAT
%
% This individual file collects data from a real pilot plant system
% which consists of two tanks and a control system to manage the
% level of the second one. This data base has been generated from
% three possible situations: normal operation (class one), a partial
% obstruction between tanks (class two), and the emergency situation,
% total obstruction at the exit of tank two (class three).
%
% Used descriptors are: level in tank one, level in tank two (both from 0 to
% 10), the desired output or control input (only High or Low), the control
% signal (also from 0 to 10), and the derivatives of first, second level and
% control signal (coded: Negative Large, Negative Small, Zero,
% Positive Small and Positive Large).
%
% We do have a last field of imposed class.

&LEVEL_T1 LEVEL_T2 OUTPUT CTRL_SIG DERIV_T1 DERIV_T2 DERIV_CONTROL

9.107939 3.021866 A 5.622508 D D C 1
9.312390 2.988349 A 5.940937 B C D 1
7.678884 1.027721 C 4.984652 A B D 1
9.994041 2.507010 A 4.953032 D A E 1
5.344999 2.015358 C 3.481344 E E A 1
9.031239 3.020050 A 5.714987 D D B 0
8.579471 1.061223 C 4.925020 D B D 2
8.196658 0.961353 C 6.249512 E D B 2
8.619834 0.909940 C 6.692107 A D B 0
7.391809 0.987452 C 5.994978 E D B 2
8.945312 3.056641 A 5.458984 B D B 1
6.353343 5.844610 A -1.251475 E A C 1
5.928058 6.129644 A -1.275735 C C C 3
5.927319 6.129463 A -1.276346 D C C 3
5.955299 6.138647 A -1.276663 C C C 0
5.958092 6.142730 A -1.283172 D C C 3
7.255859 0.976562 C 5.537109 D B C 1
```

FIGURE 3 – EXAMPLE OF THE DATA (.DAT) FILE

3.1.1.2 Loading a training file

To load a file the user must select either the ‘**New Context**’ or the ‘**New Population**’ buttons (Figure 4). Although it is possible to start with any of the two options it is recommended to first load the context data file and then the population file.



FIGURE 4 – NEW CONTEXT & NEW POPULATION BUTTONS

- **Loading a new context:** The context file normally contains the number of descriptors and its type (numeric or symbolic). It may be induced from a population (or plain text) file with a .DAT extension or retrieved from a binary file previously saved with a .CON extension (Figure 5). If the context has been ‘induced’ from a population the application will demand the user if he wants also to load the corresponding population to be used for classification.

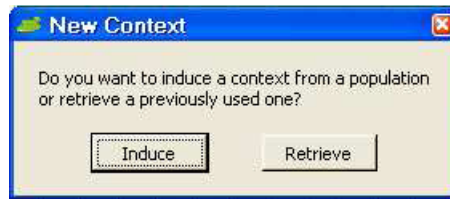


FIGURE 5 - LOADING A NEW CONTEXT: INDUCE OR RETRIEVE

- **Loading a new population:** This file contains the individuals that will be used for training. It must have the same number and type of descriptors as in the context and it may contain an additional field corresponding to the pre-defined classes (to be used for the supervised learning case). A dialog window will appear asking the user if he wants to load the corresponding context (Figure 6). If changes have previously been made (e.g.: a descriptor has been disabled) to the context, they will be lost if the user accepts loading the new context also.

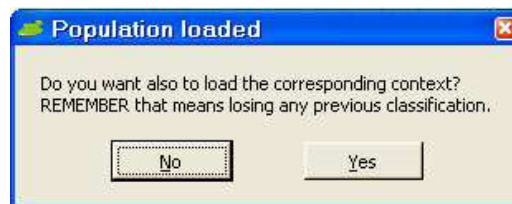


FIGURE 6 – LOADING A 'NEW' POPULATION

Because of the presence of commented lines (with a '%' symbol) and a possible supervised learning field, there exist a couple of dialog windows that ask the user if the population file is to be looked at or edited with a Windows text editor, and if the last field is to be treated like an imposed classification value. See Figure 7.

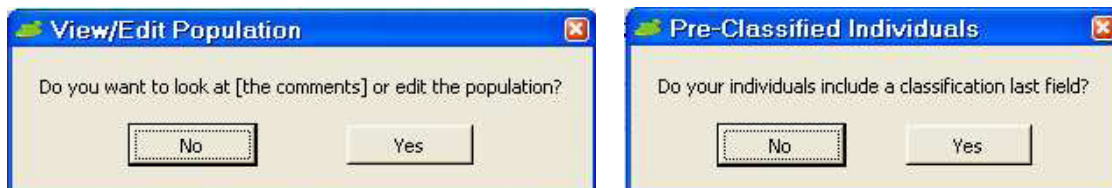


FIGURE 7 - VIEW/EDIT FILE & IMPOSED CLASSIFICATION WINDOWS

3.1.2 Choosing the Classification Algorithm - Quantitative descriptors

To make the different calculations required in the learning and classification procedures it is necessary to choose the type of algorithm to be used for the quantitative descriptors (Figure 8):

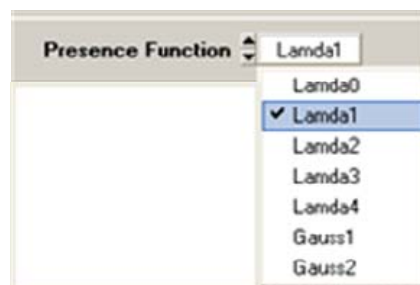


FIGURE 8 - PRESENCE FUNCTION SELECTION

$$\text{Lamda1} : \rho^x (1-\rho)^{1-x}$$

$$\text{Lamda2} : K \rho^x (1-\rho)^{1-x} \quad K = \frac{\log\left(\frac{\rho}{1-\rho}\right)}{2\rho-1}$$

$$\text{Lamda3} : \rho^{1-|x-c|} (1-\rho)^{|x-c|}$$

$$\text{Lamda4} : K \rho^{1-|x-c|} (1-\rho)^{|x-c|} \quad K = \frac{\log\left(\frac{\rho}{\rho-1}\right)}{2\rho - \rho^c (1-\rho)^{1-c} - \rho^{1-c} (1-\rho)^c}$$

$$\text{Gauss1} : e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\text{Gauss2} : K e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad \text{with adjustable minimum threshold (GAD NIC)}$$

3.1.3 Logic operators and the exigency level

3.1.3.1 Logic operators

The user may choose between two different families of logic operators (T-norms & dual T-conorms functions) in order to aggregate all the Marginal Adequacy Degrees of an individual to a class (Figure 9). These operators are: Product-Probabilistic Sum (T-norm & T-conorm) and the Minimum-Maximum (T-norm & T-conorm).

3.1.3.2 Exigency Level

The user may also use mixed connectives of the same family by choosing an exigency level (α) between 0 and 1 (Figure 9). In recognition we call α the *exigency level* and in self-learning it is called *selectivity level*. By changing the value of α different partitions based on the same data used may be obtained. Thus, as the value of α increases more classes will be created or in the case of recognition a greater adequacy is required of a measurement to be assigned to a pre-established class.



FIGURE 9 – CONNECTIVES & EXIGENCY SELECTION

3.1.4 Classification mode

When a file has been loaded for the first time the classification mode will be selected according to the characteristics of the file.



FIGURE 10 – CLASSIFICATION MODE SELECTION

3.1.4.1 Self – Learning

Also known as unsupervised learning, meaning that there are no pre-defined classes (no preliminary knowledge of how the elements are assigned). SALSA creates a group of classes using as base the values of the descriptors from the individuals of the data file that has been loaded.

FIGURE 11 – SELF-LEARNING SELECTORS

The **‘Manual / Automatic’** selector (Figure 11) gives the possibility to make an automatic scanning, combining the three principal parameters of LAMDA algorithm. The result is a table with the number of classes generated by LAMDA using self-learning and the partition’s quality, given by the quality index CV proposed by (Isaza, 2007) – see Figure 12 . The partition with the highest quality value for each MAD function is highlighted (number of classes generated and CV index value).

Automatic Classes

Partition Quality

For each Presence Function the classification with the highest Quality is highlighted

	EXIGENCY	LAMDA 1	LAMDA 3	GAUSS 1
	0.2	2	1	1
	0.4	2	1	1
MIN-MAX	0.6	2	1	1
	0.8	26	32	15
	1.0	35	35	36
	0.2	2	2	1
	0.4	2	2	1
PROBABILISTIC	0.6	2	2	1
	0.8	2	2	1
	1.0	5	6	6

Automatic Classes

Partition Quality

For each Presence Function the classification with the highest Quality is highlighted

	EXIGENCY	LAMDA 1	LAMDA 3	GAUSS 1
	0.2	0.0160281	0.0168595	0.0167566
	0.4	0.0157978	0.0163123	0.0162594
MIN-MAX	0.6	0.0150315	0.0153416	0.0153378
	0.8	0.0625193	0.0902129	0.0591460
	1.0	0.1170478	0.1215640	0.1280458
	0.2	0.0008145	0.0005536	0.0172148
	0.4	0.0008362	0.0006042	0.0171556
PROBABILISTIC	0.6	0.0008794	0.0007021	0.0170087
	0.8	0.0010069	0.0009750	0.0165120
	1.0	0.0197041	0.0180866	0.0153527

FIGURE 12 - AUTOMATIC SELF-LEARNING RESULTS: CLASSES & PARTITION QUALITY

The other two selectors in Figure 11 that may be changed by the user when unsupervised learning has been selected are the maximum desired variation percentage (*‘Max. Variability’*) and the maximum allowed iterations number (*‘Max. Iterations’*). They are necessary because in unsupervised learning the class parameters vary, so to obtain some stability and to overcome in a certain way the effect of the observations ordering, the same population is classified several times until a maximum percentage of individuals changes from one iteration to another. However the time to reach stability could be very long, so a maximum number of iterations have also been introduced. This means that the population will be presented to the existing classes once and again until either no more than the specified percentage of individuals varies its assignment from the previous iteration or the cycles limit has been reached. In either case the total iterations made by the algorithm is displayed (see *‘Total Iterations’* in Figure 11).

3.1.4.2 Active Supervised Learning

The loaded population has a class imposition field. This option allows performing a different number of choices, like learning from an initial set of classes, which can be modified by adding new classes or by updating their parameters or both (Figure 13).



FIGURE 13 – SUPERVISED LEARNING OPTIONS WINDOW

3.1.4.3 Pattern Recognition

This option has two alternatives, either the user allows unclassified individuals, meaning that an individual has not been recognized in any class (its adequacy degree is lower than the minimum threshold) and has been placed in the NIC class; or force every individual to be assign to a class. In this last case the Non-Informative Class is not taken into account for recognition (Figure 14).

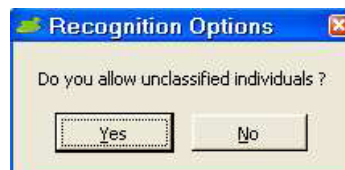


FIGURE 14 – RECOGNITION OPTIONS WINDOW

3.1.5 Data Processing, Treatment and Visualization

- Once the context and the population have been loaded, SALSA identifies the number of descriptors and its nature, and it internally rescales quantitative values between 0 and 1 (“Normalisation” process).
- To make possible a direct confrontation between individuals and classes, it is necessary that the descriptors used for observations are of the same type and in the same order than the concepts used for classes (context). However, whenever the number of descriptors, or its nature, or the actual values exhibited in the population do not fit the ones stored in the context, the problem is reported in an error file, named “... \error.out”. The error is explained in this file with an indication of its nature type and location (number of individual and number of descriptor if necessary). A message window is displayed to notify the user. If the population can still be loaded, for instance, if a value is lower than the minimum value, SALSA will take into account a 0; or if a value is greater than the maximum value SALSA will replace it with a 1; and the process will continue.
- The user may view each context descriptor separately and in a detailed way, as well as the values of each individual (Figure 15). It is possible for the user to change the descriptors he wants to use for learning and classification; he can *enable* and *disable* a descriptor for classification.

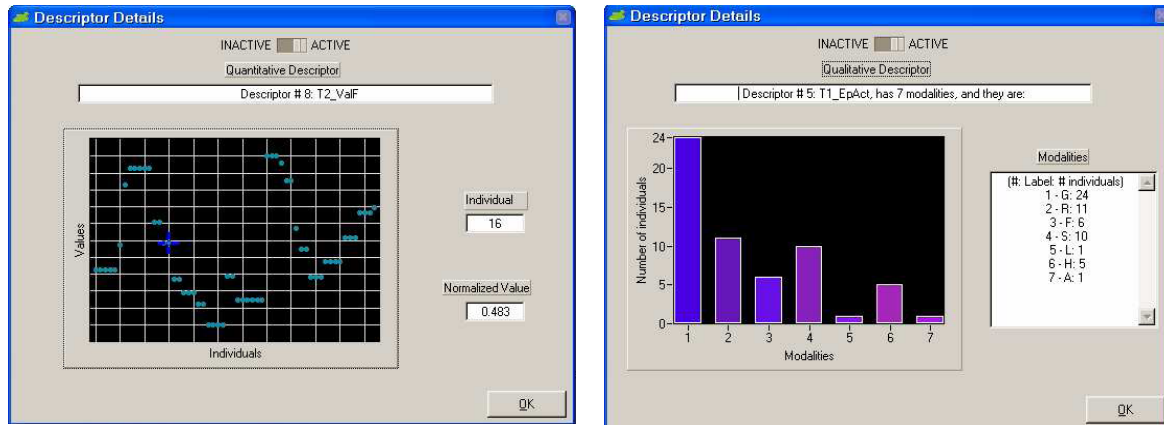


FIGURE 15 - QUANTITATIVE & QUALITATIVE DESCRIPTOR WINDOWS

For qualitative descriptors, a histogram of the number of individuals for each modality is shown in Figure 15 as well as a small window that gives the name of the modality and the exact number of individuals. For quantitative descriptors a graph representing each individual with its normalized value is shown. There is also a text window with all the information about the loaded context (see Figure 16).

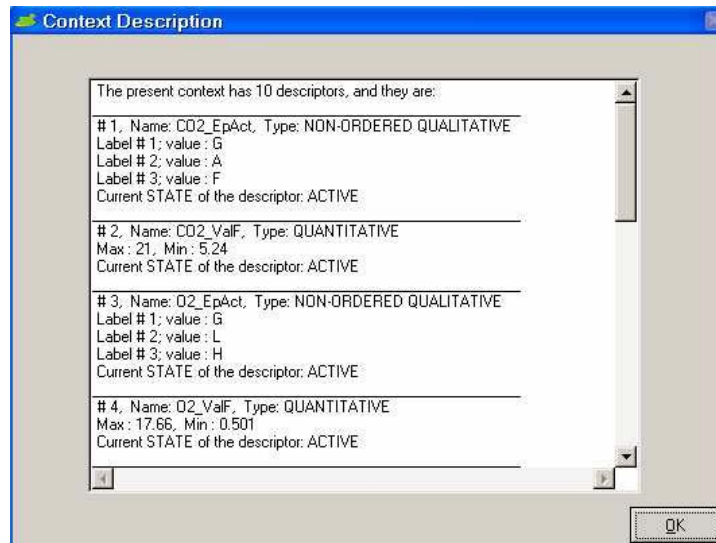


FIGURE 16 – CONTEXT DESCRIPTION WINDOW

By clicking the **'View Population'** button all active descriptors are graphically displayed on the lower graph of the main screen. There is a graph for quantitative descriptors where for each individual its normalized value is shown, and at the right part of the graph the names of the active quantitative descriptors are displayed. For qualitative descriptors, modalities corresponding to each individual are shown in a graph. The user may change from one graph to another using the corresponding option: 'View Qualitative Descriptors' or 'View Quantitative Descriptors', which is located at the right bottom corner of each graph. (See Figure 17 & Figure 18). For qualitative descriptors we represent each modality with a similar but different colour.

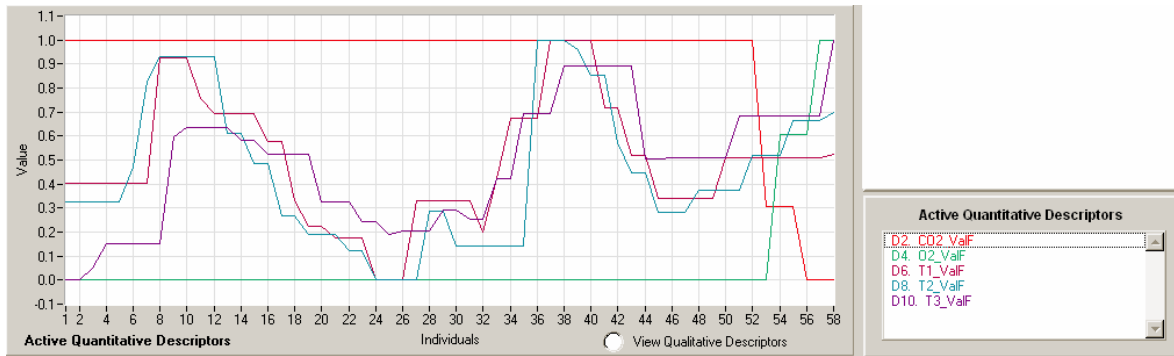


FIGURE 17 – ACTIVE QUANTITATIVE DESCRIPTORS GRAPH

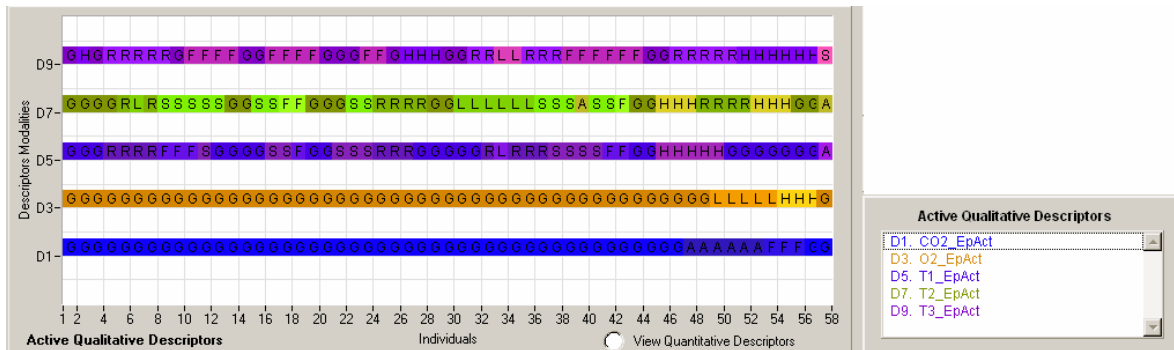


FIGURE 18 – ACTIVE QUALITATIVE DESCRIPTORS GRAPH

3.1.6 Classification Results and Visualization

After successfully loading a context and population (green LEDs for Context and Population and name of the loaded file), it is possible to put the algorithm to work by pressing the **CLASSIFY** button (see Figure 19).

CONTEXT

New View Save

ATA_TEST\Gazeifieur_Episodes-Apprent_5V2D.dat

POPULATION

New View Save

ATA_TEST\Gazeifieur_Episodes-Apprent_5V2D.dat

CLASSIFICATION

Retrieve Save

<not saved>

Presence Function

Gauss1

Connectives

Probabilistic

Exigency Index

1.000000

Classification Mode

Self - Learning

CLASSIFY

Class Profile

Save Ref.Cl

Text

Class Details

View Ref.Cl

Analyse

FIGURE 19 – APPLICATION READY FOR CLASSIFICATION

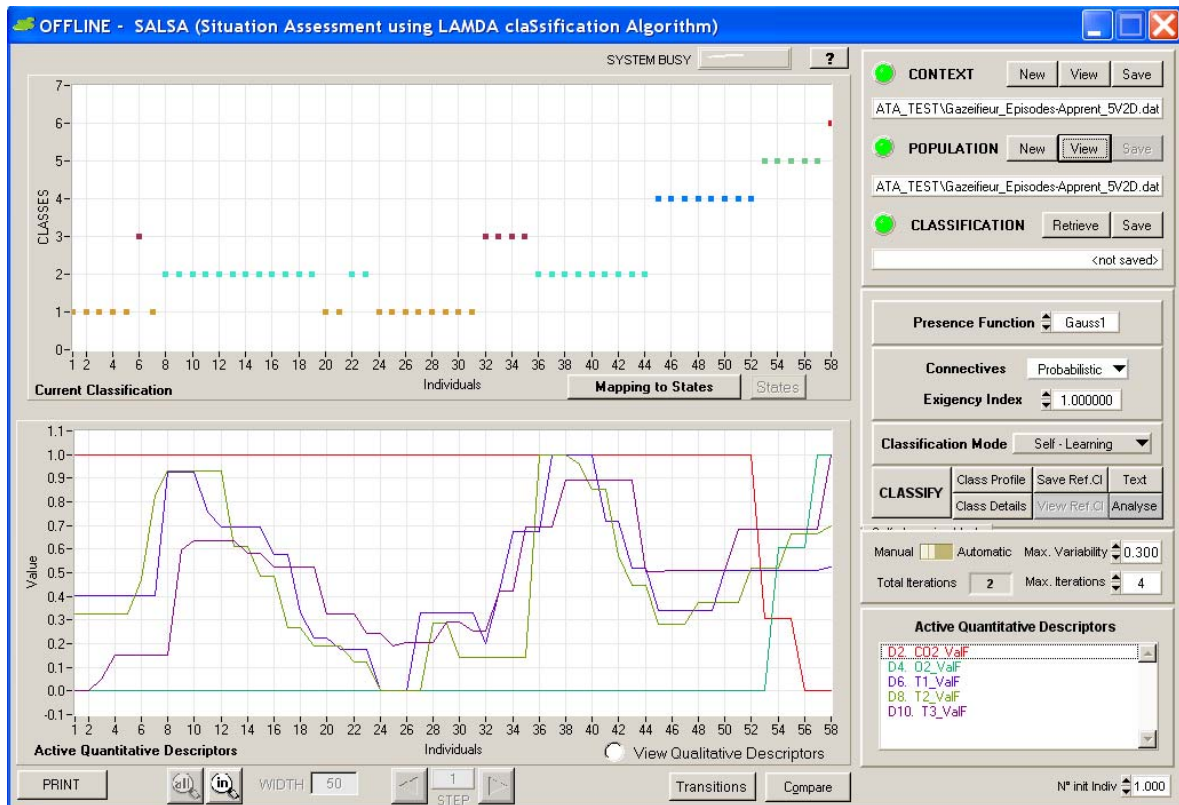


FIGURE 20 – CLASSIFICATION RESULTS USING SUPERVISED LEARNING

3.1.6.1 Resulting classes - characteristics

Once a classification has been made, SALSA graphically displays the class assigned to each individual (Figure 20), and by clicking on the **Class Profile** button (Figure 21) the profile of each class will be displayed.



FIGURE 21 – CLASSIFICATION VISUALIZING BUTTONS

Figure 22 shows the profile for each class. Each descriptor is represented with a different colour, for quantitative descriptors their mean value is displayed, for qualitative descriptors each modality is represented with a different colour, and their occurrence frequency is shown (the sum of all modalities for a qualitative descriptor must be 1).

It is also possible to view the complete information for each class in text mode (**Class details** button - Figure 21). The Class Details window shows in a table all the parameters values for each class. Also the number of elements (individuals) assigned to each class is displayed (Figure 23). Depending on the type of algorithm used for classification different information will appear. The user may select to view and export (into a CSV file) the class parameters either in a normalized format or with its absolute values.

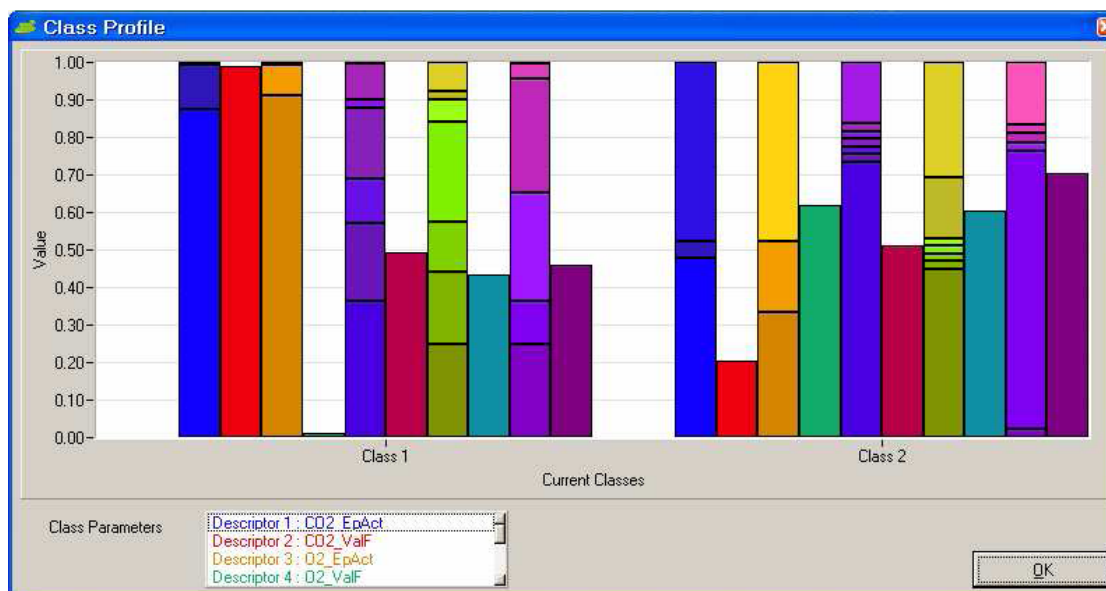


FIGURE 22 – PROFILE FOR EACH CLASS

Normalized Values		Absolute Values								
	Number of Elements	D1 CO2_EpAct-C	D1 CO2_EpAct-A	D1 CO2_EpAct-F	D2 CO2_ValF-Freq	D3 O2_EpAct-C	D3 O2_EpAct-L	D3 O2_EpAct-H	D4 O2_ValF-Freq	D4 O2_ValF-Freq
CL 1	39	39	0	0	20.8124	39	0	0	0.7053	
CL 2	5	5	0	0	19.0300	5	0	0	2.6459	
CL 3	8	2	6	0	20.1244	4	4	0	1.4543	
CL 4	5	2	0	3	8.9633	1	1	3	9.9902	
CL 5	1	1	0	0	9.1800	1	0	0	13.3702	

FIGURE 23 – CLASS DETAILS WINDOW

3.1.6.2 Recognition Results

When Recognition is launched, and if the loaded population has pre-defined classes a 'Results Window' is displayed with different information (Figure 24): number (and percentage) of individuals not recognised, number of individuals with pre-defined classes, number (and percentage) of individuals badly recognised, and the list of the individuals badly recognised.

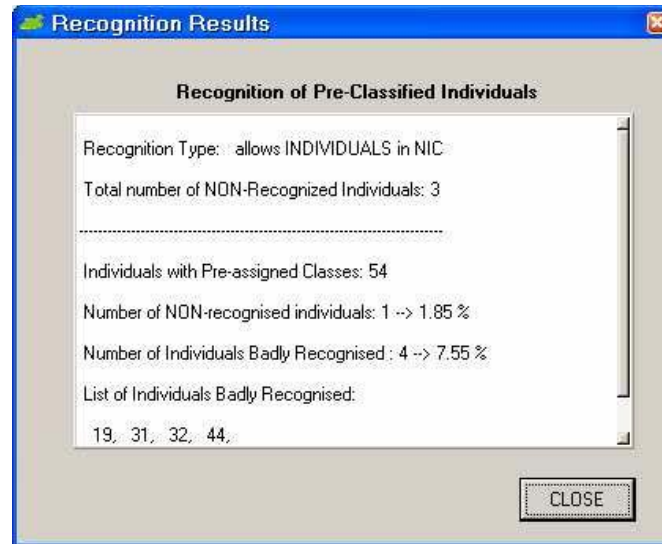


FIGURE 24 - RECOGNITION RESULTS WINDOW

When a population with pre-defined classes has been loaded the **'View Prof CL'** button is enabled (Figure 21). By clicking this button the pre-defined classes for each individual are displayed (Figure 25).

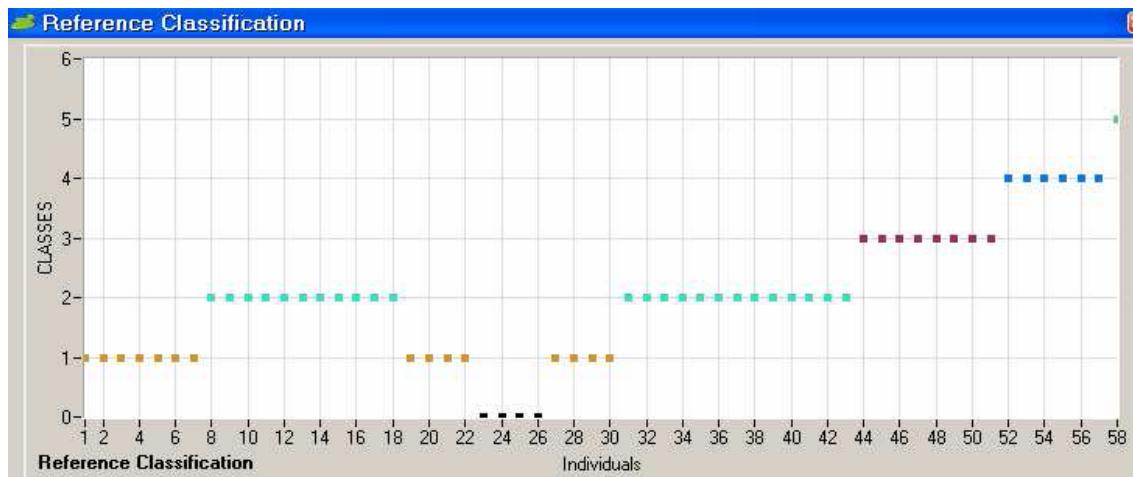


FIGURE 25 - REFERENCE CLASSIFICATION GRAPH

SALSA automatically creates a result file where each individual has its corresponding class. This file is a text file and has a .RES extension. It is possible to **save** the classification result as the 'reference' one by clicking on the **'Save Ref. CL'** button. The user may choose the name of the classification, and he can decide to set it as the reference for comparison with other classification results.

3.1.6.3 Classification Analysis

The **'ANALYSE'** button shows the resulting membership matrix of each individual for each existing class (Figure 26). This matrix can also be graphically viewed by clicking on the **View Membership Graph** button (Figure 27). The user may save this matrix in a *.CSV format by clicking on the **EXPORT** button. The analysis window displays as well the partition quality index value, which allows the user to make a comparison between other classifications. The higher the value of the CV index, better the partition's quality.

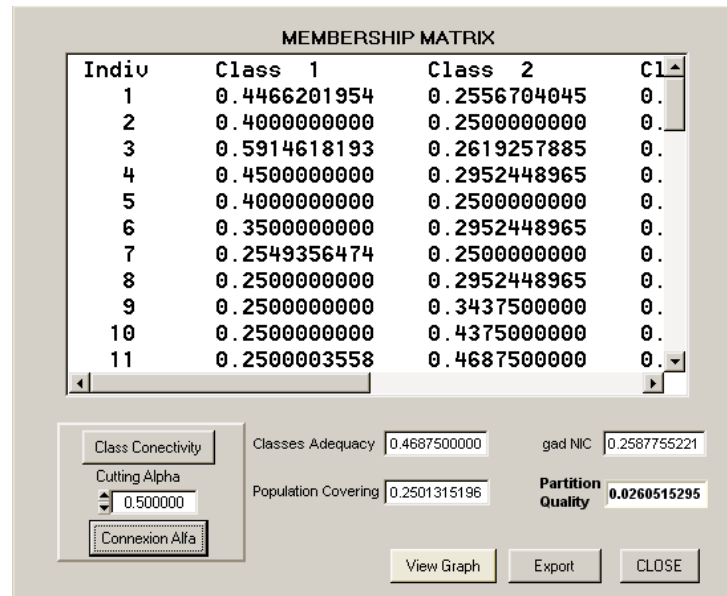


FIGURE 26 - ANALYSE WINDOW

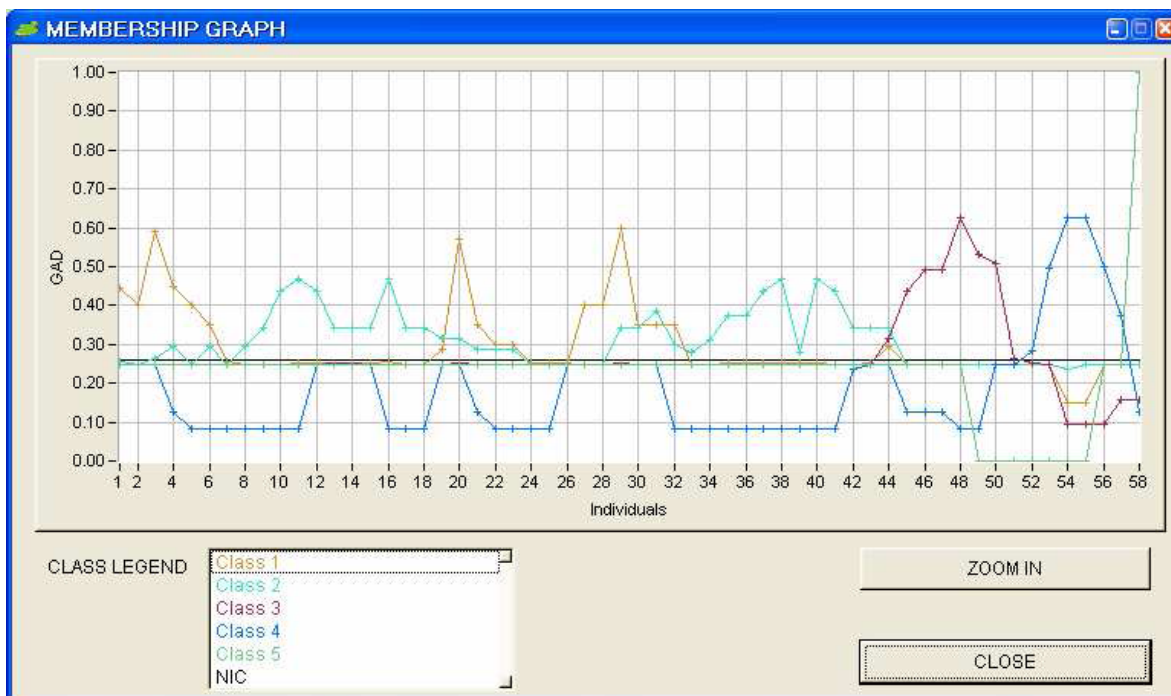


FIGURE 27 - MEMBERSHIP GRAPH

3.1.7 Mapping Classes to States

When a suitable classification is obtained the user must assign the different classes into representative states. This is done using the **'Mapping to States'** button, located at the right bottom side of the 'Current Classification' graph.

The user must first of all create the list of possible significant states, by clicking at the 'Create List of States' button (see Figure 28).



FIGURE 28 - MAPPING CLASSES TO STATES WINDOW

The user can add and remove the number of representative states of the process. Once the list is completed the user can close the window by clicking the OK button (see Figure 29).



FIGURE 29 - CREATING THE LIST OF STATES

After the list of states is created the table of classes and states must be filled up. A class can be mapped into a state from the list by double-clicking into the cell of the corresponding class (Figure 30).

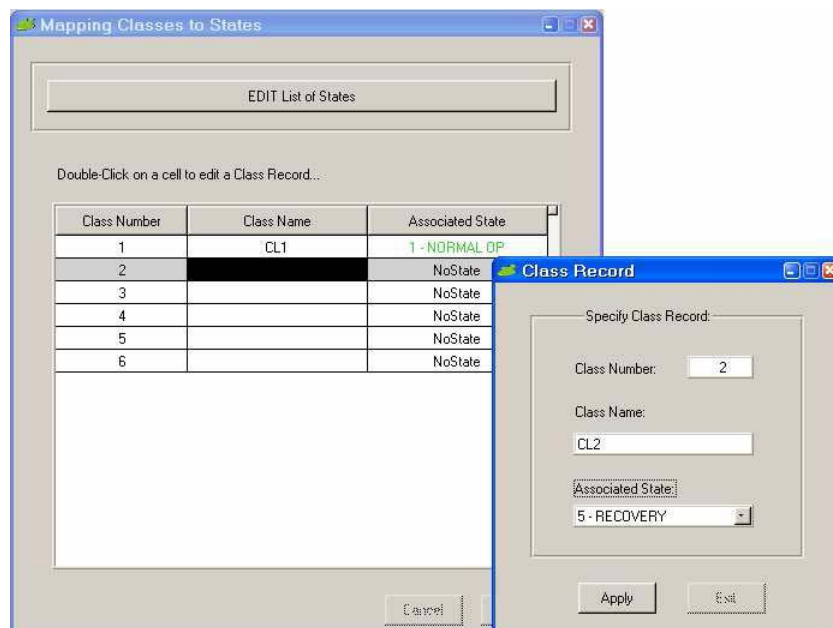


FIGURE 30 - ASSIGNMENT OF CLASS TO STATE

The table may be viewed or edited with the **"View/Edit Mapping"** button. The user can also view the assignment of states to the individuals by clicking the **States** button (Figure 31).



FIGURE 31 - VIEW/EDIT MAPPING & STATES BUTTONS

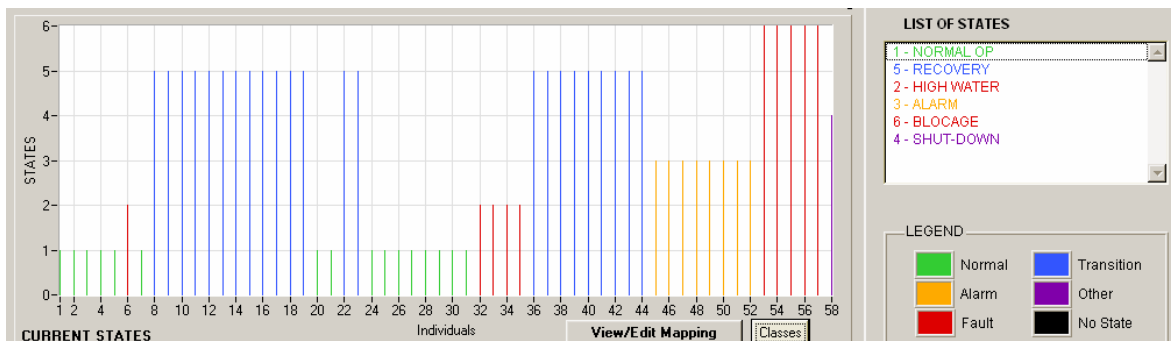


FIGURE 32 - CURRENT STATES GRAPH

SALSA allows saving the different contexts used (active descriptors, presence function for quantitative descriptors, connectives and exigency level), as well as the classification results for future use. The context will be saved in a binary format with a .CON extension. Whereas, classification parameters (classes and states) will be saved within a binary format with a .CLA extension (See Figure 19 for location of **save** buttons for context and classification).

3.2 SALSA – ONLINE

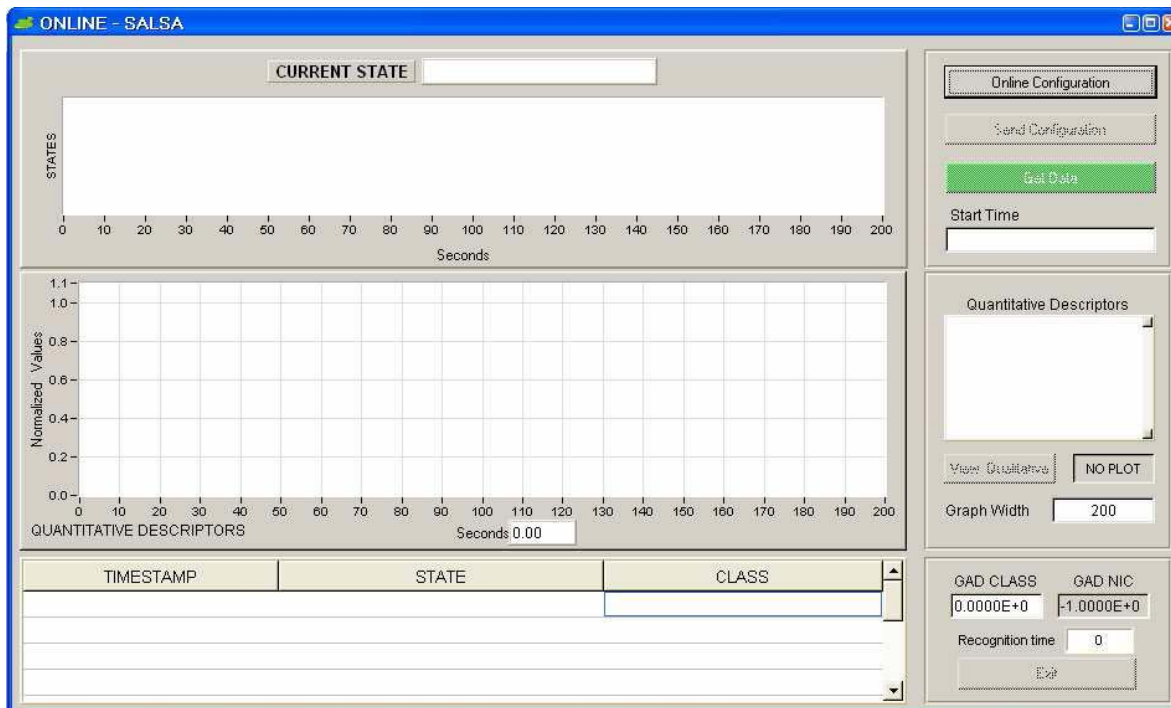


FIGURE 33 - SALSA ONLINE MAIN SCREEN

These are the steps the user must follow for online recognition.

3.2.1 Online Configuration

The online configuration button is the only enabled button when the user chooses the online stage of SALSA in the main menu (Figure 33).

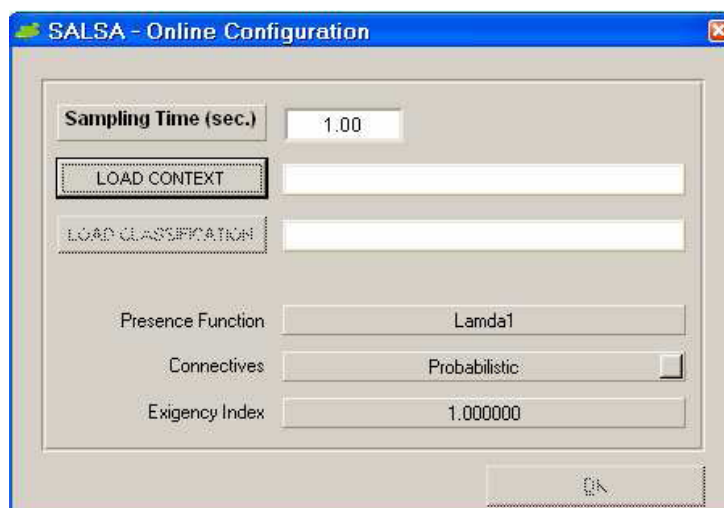


FIGURE 34 - SALSA ONLINE CONFIGURATION

- The user may adjust the sampling time for the online recognition stage. The possible values go from every 0.5 seconds up to every 5 minutes.
- The user must load up the context that will be used for recognition, i.e. the number and type of descriptors used on the offline stage for the creation of classes (Figure 34). This file is a binary file with a .CON extension (this file should have been previously created during the offline stage).
- Once the context is loaded the user must choose a classification file with .CLA extension. This classification must have the same number and type of descriptors as the loaded context. Once the user has loaded the classification, the learning parameters used for this classification will appear under the labels: “Presence Function”, “Connectives”, and “Exigency Index”. Once both files have been loaded the user may close the configuration window by clicking on the **OK** button.

3.2.2 Online Connection

Once the configuration window is closed the user will see the list of the quantitative descriptors (or the qualitative descriptors) that will be received online (see Figure 35)

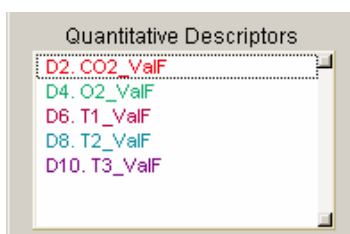


FIGURE 35 — LIST OF ONLINE QUANTITATIVE DESCRIPTORS

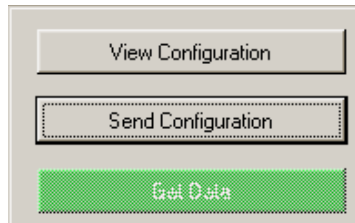


FIGURE 36 – READY TO SEND ONLINE CONFIGURATION

The 'Send Configuration' button is enabled (Figure 36). The user must click this button for SALSA to get ready to receive online data. If the connection is successful SALSA will start checking for a new message to arrive.

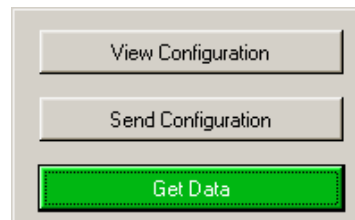


FIGURE 37 – READY TO RECEIVE ONLINE DATA

The **Get Data** button is enabled (Figure 37). SALSA will display the path where the data transfer will take place (Figure 38). The "DataTr" folder must contain the file with the current information message. This file has the name: **d_in.dat**. It contains one single line in the same format as for the training files (see Figure 39). Once SALSA reads the d_in.dat file it destroys it, and when recognition is made an output file is generated (**d_out.dat** - see Figure 40-). The application in charge of sending the online data must destroy this file and create the d_in.dat file again to tell SALSA that a new message has arrived.

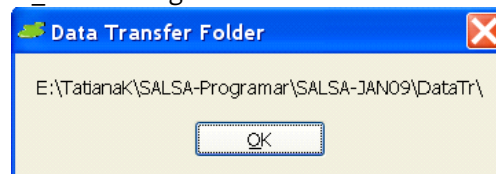


FIGURE 38 – PATH FOR DATA TRANSFER

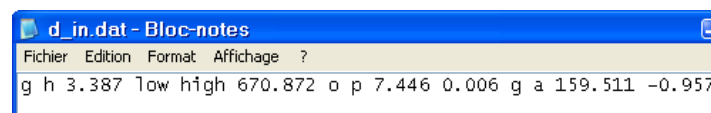


FIGURE 39 – FORMAT FOR THE ONLINE DATA FILE

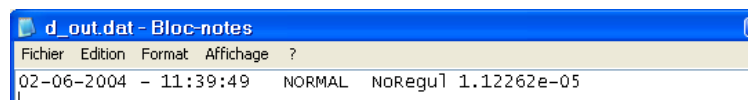


FIGURE 40 – FORMAT FOR THE OUTPUT FILE BY SALSA

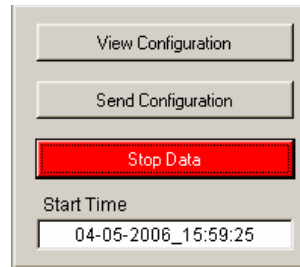


FIGURE 41 – STOP RECEIVING DATA

The user may stop receiving messages with the **Stop Data** button (Figure 41).

3.2.3 Receiving Online Data

The online screen offers the visualization of the incoming data, the current state and the state of the plant at the instance of an arriving message. It has a historical table with the time at which there was a change of state, with its associated class and corresponding GAD (*Global Adequacy Degree*).

3.2.3.1 Visualizing incoming data

The user may choose to visualise either the quantitative or the qualitative data.

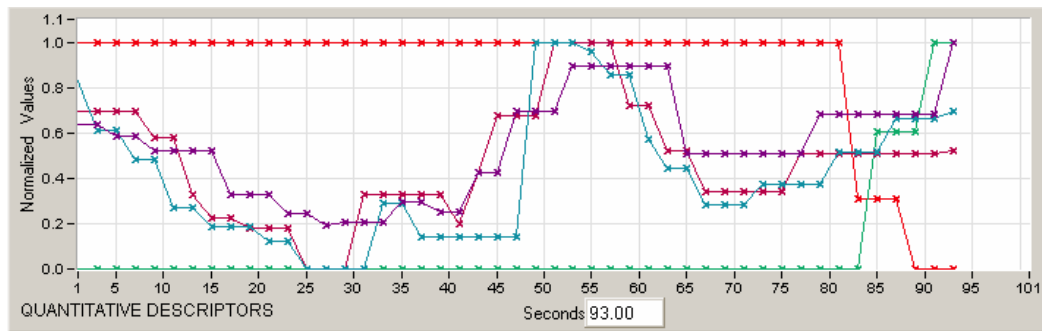


FIGURE 42 - ONLINE QUANTITATIVE DESCRIPTORS

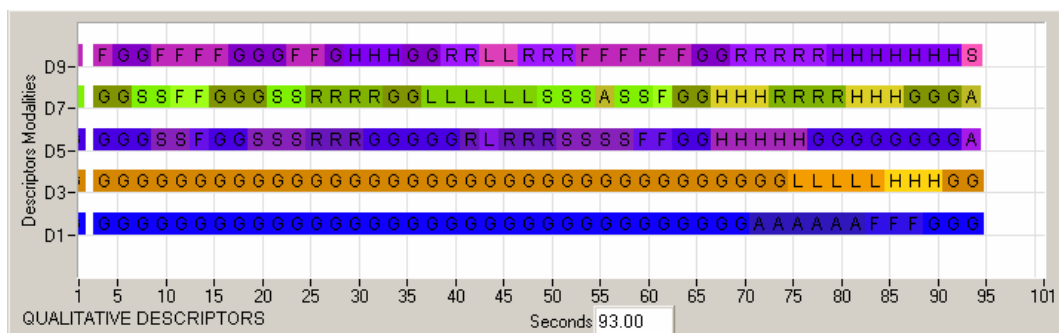


FIGURE 43 - ONLINE QUALITATIVE DESCRIPTORS

3.2.3.2 Visualizing the Current State

- The user may visualise the current state of the plant at the top of the screen: a bar graph indicating the state of the process each time a new observation arrives (Figure 44).

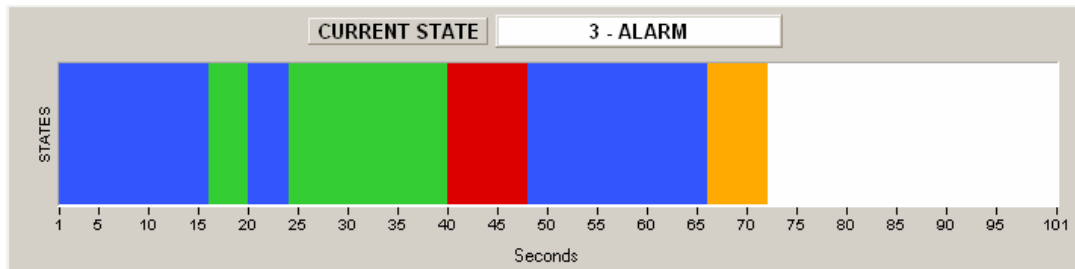


FIGURE 44 - BAR GRAPH WITH STATES

- A Table with the timestamp of a change of state, the state and its associated class for the last 10 states is displayed. The current state is highlighted with the corresponding colour of the state (Figure 45):

TIMESTAMP	STATE	CLASS
04-05-2006 - 16:23:06	3 - ALARM	CL4
04-05-2006 - 16:22:48	5 - RECOVERY	CL2
04-05-2006 - 16:22:40	2 - HIGH WATER	CL3
04-05-2006 - 16:22:24	1 - NORMAL OP	CL1

FIGURE 45 - TABLE WITH CHANGE OF STATE OCCURRENCES

To EXIT the online screen the user must **FIRST STOP** the online connection with the **STOP** button and then **EXIT** the screen with the EXIT button.

4. APPENDIX

4.1 ONLINE DATA GENERATOR

This application permits to generate online data from stored data files. It can be used in order to validate SALSA Online stage.

The Online Data Generator is a standalone executable that may be installed in the following way:
In the InstallationSALSA04v1 Folder look for the sub-folder: Package. Click the 'setup.exe' file.

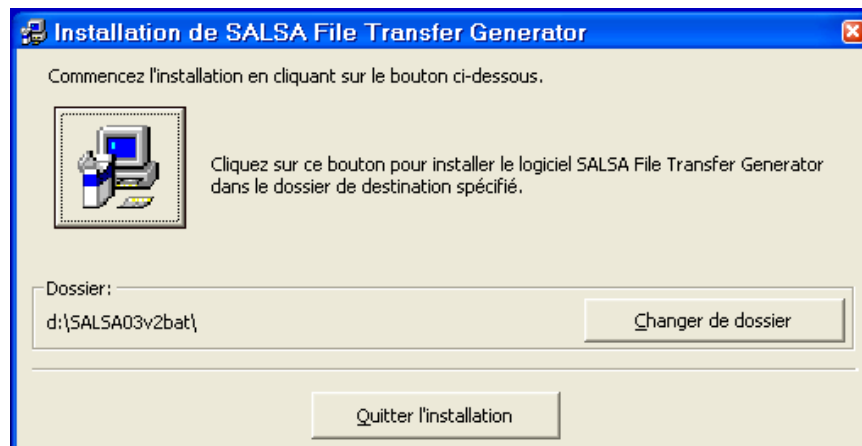


FIGURE 46 - ONLINE DATA GENERATOR SETUP WINDOW

Once the installation is completed it is possible to generate online data from a file.

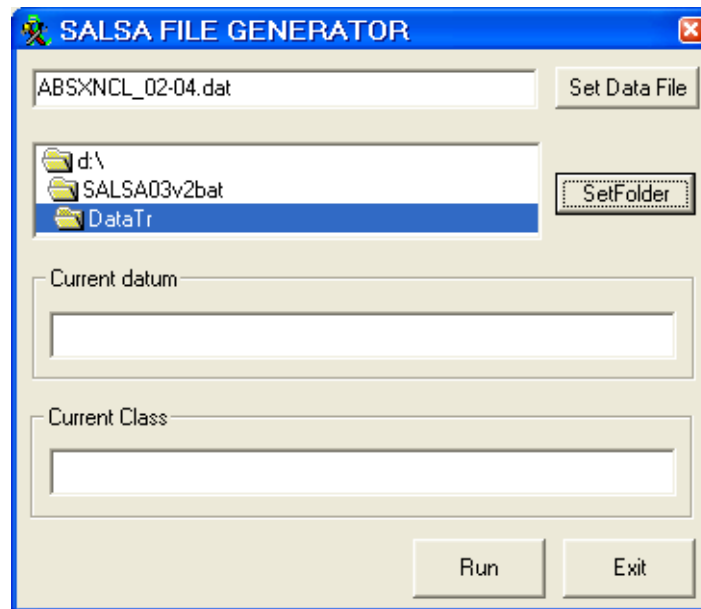


FIGURE 47 - ONLINE DATA GENERATOR MAIN SCREEN

First of all choose the data file with extension *.DAT and click the **Set Data File** button (see Figure 47). Choose the folder where the online data file must be created (it must be in the \DataTr folder created by SALSA: ... \SALSA04v1 \DataTr) and click the **Set Folder** button. The RUN button is enabled. When clicking this button the program will create the **d_in.dat** file, the information of this file will be displayed in the 'current datum' window. Online Data is generated every two seconds. When SALSA creates the output file **d_out.dat** the program will display its contents in the 'Current Class' window, and will destroy this file in order to generate a new d_in.dat file with a new datum. To quit the application click on the EXIT button.

5. REFERENCES

- Aguilar-Martin, J. and López de Mántaras, R. (1982).** The Process of Classification and Learning the Meaning of Linguistic Concepts. In: *Approximative Reasoning in Decision Analysis*. p. 165-175.
- Isaza, C. (2007).** *Diagnostic par techniques d'apprentissage floues : conception d'une méthode de validation et d'optimisation des partitions.*, Ph.D. dissertation. INSA
- Kempowsky, T., Aguilar-Martin, J., Le Lann, M. and Subias, A. (2002).** Learning Methodology for a Supervision System using LAMDA Classification Method. In *IBERAMIA'02-VIII Iberoamerican Conference on Artificial Intelligence*. Sevilla, Spain.
- Kempowsky, T., Aguilar-Martin, J., Subias, A. and Le Lann, M. (2003).** Classification Tool based on Interactivity between Expertise and Self-learning Techniques. In *IFAC - Safeprocess*. Washington D.C., USA.
- Kempowsky, T. (2004).** *Surveillance de procédés à base de méthodes de classification: conception d'un outil d'aide pour la détection et le diagnostic des défaillances*, Ph.D. Dissertation. INSA